

Workflow

I am sometimes asked why it is possible for me to work so concentrated and professional on my source code. A total of 9 rules have been established in recent years, which have proved useful (to me). I describe here the work for private projects, not for your company or other jobs.

1.) The 10 Minute Rule

Before I get to work, I look for e-mails again if something important has arrived. Ten minutes at the most. I don't read the mails that say *important*, because most of the time they are rather unimportant and waste my time. I only write back once a day, mostly in the middle of the night, so that I don't disturb other people at work and they can read the mails the next morning in peace. Of course I'm not a manager now and I can take advantage of this freedom, but I still see it in many people in higher positions that they respond to any bullshit. This in turn creates stress and leads to a worse workflow, so I turn off my email program when I start programming.

2.) The "No" Rule

No! You do not need a social network if you are programming. No! You don't have to be on a Mastodon! You don't have to reply to a comment at Instashit and news.ycombinator.com will be fine for a few hours without you. Log out of all social networks and if you are already there you can also cancel all accounts forever. I promise you that your projects will then run much better because you can concentrate better. I've watched developers at conventions as many times as they switch to Twitter every ten minutes to post, answer or read something in their browser. You're wasting your time and your programming code automatically gets worse and you're too easily distracted.

3.) Riddler's Rule

If, for example, I program in Python, it can happen that I simply cannot get any further at one point. The temptation is then easy to read that very quickly on the Internet. Unfortunately you don't find the important information as fast as you want, surf through 42 tabs and gradually forget what you wanted to find. If you want to program, get the required documentation as an offline version. Of course it is more convenient to look at Stackoverflow and copy the solution. But you only cheat yourself, because you don't learn anything from it. But if you have to work hard on the puzzle yourself, you will keep the learned information much better. If you really can't get any further at one point, you simply don't bother enough, because there are no unsolvable programming problems.

4.) Bell's Rule

We do have a home phone, but it's usually only used by the authorities or the health insurance company. I [don't have a smartphone](#) and if I had one I would turn it off while programming. Nobody has to disturb you in your work. For most people it doesn't help to mute the phone because they always check the screen every few minutes to see if someone has called. My philosophy is *If your messages are important enough, they will reach me*. Nobody has to be able to reach me by phone and should someone from my family die, the police will ring my doorbell after a while. Dead people don't care if I show up later.

5.) The Gothic Rule

The night is your friend. During the day I do my household, at 15:00 I take a nap for one hour, afterwards I write articles or do things for the authorities. At 21:00 o'clock I go to sleep and get up again at 23-24:00 o'clock. Then program the night through until 6 o'clock in the morning and then go back to sleep until 9:00 o'clock. At night no people come to bring my ordered parcels, the neighbours all sleep and there are no children, cars or craftsmen on the road. Then the world is a little calmer in my selective perception and peace is a very important factor to get ahead in my project.

6.) The Master Craftsman Rule

My grandmother was a trained master craftswoman and my mother had also completed her education as a blacksmith before she trained as a technical drafter. What I learned from these two people is that the tool is crucial for the work of a master. Of course you can use white glue instead of bone glue to build a violin. You can also shit on a piece of wood and call it a violin. The same comes out if you use bad tools or material for your work. This is also true for software and frameworks. If programs keep crashing, causing problems, or don't install properly, you don't come to programming. I personally prefer software from free projects because I am used to fixing bugs myself. But I can't give a general recommendation, because everyone has his own style.

7.) The Liquid Wizard Rule

Before I program I drink a small bottle (0,5l) of water. No lemonade, fake juices, cola, cofee or

energy drinks because there's too much sugar in it and I don't need to be a nutritionist to know it's unhealthy for your body. If you want to make some juice, buy some fruit and put it in the blender. Especially in Silicon Valley, there seem to be people who think it takes too much time. All I can say is that if you don't have time to mix a juice before a programming session, your schedule is already completely wrong at your base. Water has proven to be the best alternative so far.

8.) The 30/5 Rule

I don't really take a long break while programming because I'm mostly in the tunnel. Nevertheless, I get up every 30 minutes and walk around the room a little, stretch my legs or open the window to let fresh air into the room. It also helps me to relax my brain and think about something else. I personally don't know anyone who can work on a project for twelve hours at a time without being unfocused in between. Apart from scientists on the International Space Station, but they also have completely different prerequisites. All in all, you should take a break at least once an hour. At this point I have to admit that it is really difficult to keep to this rule.

9.) The whispering in the Cloud Rule

As I write this article I watch a Twitch stream and can still concentrate on the content of the article. Since I have already written down all the key points during the week, I just have to summarize my notes in sentences and I have to think about it. When programming I do without radio because too many interruptions, *news* and advertisements disturb my concentration. At Bandcamp or music playlists I have to concentrate on the list or move on to the next track etc. That also disturbs me.